This document contains information about laboratory exercises.

Object programming - classes 10

Subjects - Null, exceptions, runtime type checking, methods *equals*() i *hashcode*()

1. Empty object - null.

1. Download and import to eclipse project for today laboratories.
2. Create new class and define 2 variables, first of type *String* and second of type int.
3. Write to the console values of objects created in point *b*). What are the default values of objects and variables ? Can we set object value to null ? How check if an object is not-null ?
4. What happen if we invoke method on object that have assign null value ?

2. Exceptions

1. In package Exceptions create new class A extending class Exception.
2. In class Exception, in method *two*() between writing to the console add condition, when variable *isException* is equals *true* throw an exception A().
3. Correct error by catching throwed exception method *one*.
4. Add to *try* and *catch* block *finally*. Inside write to the console communicate *"PerformingFinally"*.
5. In package Exceptions create class B which inherits from class A. Now, in class Exception when variable *isException* is equals *false* throw an exception *B*("*Exception B*") (with some string in constructor).
6. After catching exception B write on the console message carried by this exception.
7. In method *one* after catching exception throw another exception of type *RuntimeException*(). What is the difference when you change it to throwing exception of type A or *throw e* ?

3. Runtime type checking

1. Analyze program *Rzutowanie.java*.
2. Check if it is possible to cast (a− >b,b− >a,a− >Exception,b− >Exception).

4. Methods *equals*() and *hashcode*()

1. Read in Java documentation for method *equals* of class *Object*. What conditions should the function *equals*() satisfy?
2. Analyze file *Car.java*, pay attention how method *equals* looks like!
3. Write method *equals* in class *BigCar* such that it would work correctly?